

Getting Started

CoDeSys V3 DK241

IEC Programming on the IPC@CHIP[®] Family

Development Kit DK241

Order No. 563965



Table of Contents

1	INTRODUCTION	3
2	THE IEC PLATFORM CONCEPT	4
3	INSTALLING THE IEC PLATFORM.....	5
3.1	INSTALLING THE RTS - SETTING UP YOUR IPC@CHIP [®] HARDWARE	5
3.2	INSTALLING THE TSP - SETTING UP YOUR CODESYS V3 PROGRAMMING SYSTEM.....	6
3.3	DEINSTALLING A TSP	8
4	CODESYS V3.....	9
4.1	CREATING A NEW PROJECT	9
4.2	CREATE A POU	12
4.3	BUILD PROJECT	14
4.4	ESTABLISHING THE CONNECTION	16
4.5	ONLINE.....	19
4.6	ADD HARDWARE DEVICES.....	21
4.7	CREATE OWN IO DEVICE	25
APPENDIX A	ADDITIONAL INFORMATION.....	26
A.1	WHERE TO FIND MORE INFORMATION ON THE IPC@CHIP [®]	26
A.2	WHERE TO GET SUPPORT	26
A.3	ORDER NUMBERS.....	26

1 Introduction

Welcome to CoDeSysV3@CHIP, the IEC61131-3 programming environment for the IPC@CHIP[®] family. CoDeSys V3 is one of the most powerful Windows-based IEC61131-3 programming tools available for industrial controllers. The implementation of CoDeSys V3@CHIP provides the opportunity of programming in IEC61131-3 on the IPC@CHIP[®] platform, including all the IPC@CHIP[®] processor functions in hardware and software.

CoDeSys is the name of the IEC61131-3 programming environment running on a Windows PC. Developed by the company 3S – Smart Software Solutions GmbH, it is available for free. You can download the setup for the CoDeSys Programming system on <http://www.beck-ipc.com/codesysV3>.

To get your IPC@CHIP[®] CoDeSys compatible you need a respective Run Time System (RTS). The RTS will be executed on the IPC@CHIP[®]. It manages the whole communication with the CoDeSys Programming System on your Windows PC. The RTS for the IPC@CHIP[®] is also available on <http://www.beck-ipc.com/codesysV3>.

In order to have full CoDeSys V3 functionality an IPC@CHIP[®] with the **-IEC** extension is required. The IPC@CHIP[®] SC2x3-IEC series includes the required CoDeSys V3 RTS license. In case you already have a standard SC2X3 (without the **-IEC** extension), it is possible to use it for testing purposes with full functionality, only limited by time (30 min).

This document provides a step-by-step guide on how to put the IPC@CHIP[®] and CoDeSys V3 into operation and concentrates on a system level start-up. For more information on hardware and software functionality please also read the IPC@CHIP[®] Documentation and CoDeSys V3 Documentation provided by 3S.

2 The IEC Platform Concept

CoDeSys V3 is a hardware independent Programming System which can be adapted to any industrial controller. To adapt CoDeSys V3 means to create a software collection which describes and handles the hardware of the industrial controller. The software collection which adapts CoDeSys V3 to a specific IPC@CHIP[®] hardware is called the *IEC Platform*.

With CoDeSysV3@CHIP we provide a powerful and easy-to-use development kit to create IEC Platforms for your specific IPC@CHIP[®] hardware. So you can start programming your IPC@CHIP[®] in IEC61131-3 immediately.

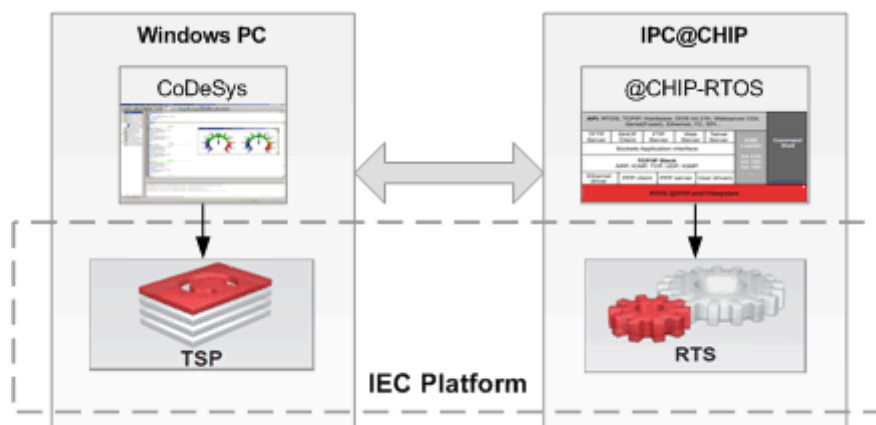
An IEC Platform is composed of two components:

1) The TSP (Target Support Package)

The TSP is a collection of files which describe hardware and software of a Target System (PLC) to the CoDeSys Programming System. Installed on your PC, it represents a Target you choose during the creation of the new CoDeSys Project.

2) The RTS (Run Time System)

The RTS is an application which runs on your IPC@CHIP[®] processor's operating system @CHIP-RTOS. It handles the communication between the IPC@CHIP[®] and the CoDeSys V3 Programming System on your Windows PC. The RTS accepts the IEC application, which is created with CoDeSys V3, and controls its execution on the IPC@CHIP[®] hardware.



The IEC Platform Concept

3 Installing the IEC Platform

Before you start working with an IEC Platform, make sure that a valid CoDeSys V3 Version is installed on your Windows PC. You can find the latest CoDeSys Setup on <http://www.beck-ipc.com/codesysV3>. For programming the IPC@CHIP[®] at least CoDeSys V3 version 3.4.3.0 Patch 1 or later is required for SC2x3 processors.

As described in the chapter before, the IEC Platform contains the RTS and the TSP. Both components have to be installed. The next two chapters describe how to install them. The RTS will be installed on the IPC@CHIP[®] whereas the TSP will be installed on your Windows PC.

For your specific IPC@CHIP[®] hardware you can create your own IEC Platform to manage the hardware and peripherals of your IPC@CHIP[®] application. You don't have to change/adapt anything at the RTS. All hardware adaptations you have to do can be done in CoDeSys V3. All hardware dependent librarys are available in CoDeSys V3. You can write your own IO driver in IEC code. You can create your own device description (TSP).

3.1 Installing the RTS - Setting up your IPC@CHIP[®] Hardware

The RTS (Run Time System) is an executable file for the IPC@CHIP[®]. Installing the RTS on your IPC@CHIP[®] means to download it to the IPC@CHIP[®] processor's flash and start it. After that your IPC@CHIP[®] is accessible by CoDeSys V3 and ready to run IEC61131-3 applications.

Before you can install the RTS on your IPC@CHIP[®], you have to set up the communication with the PC. To do that, have a look into the Getting Started Manual for the IPC@CHIP[®] Embedded Web Controller Family (<http://www.beck-ipc.com/en/products/sc2x3/sc243.asp#ql4>). There you will find out how to set up the communication, and download a program.

The Runtime directory contains the following files:

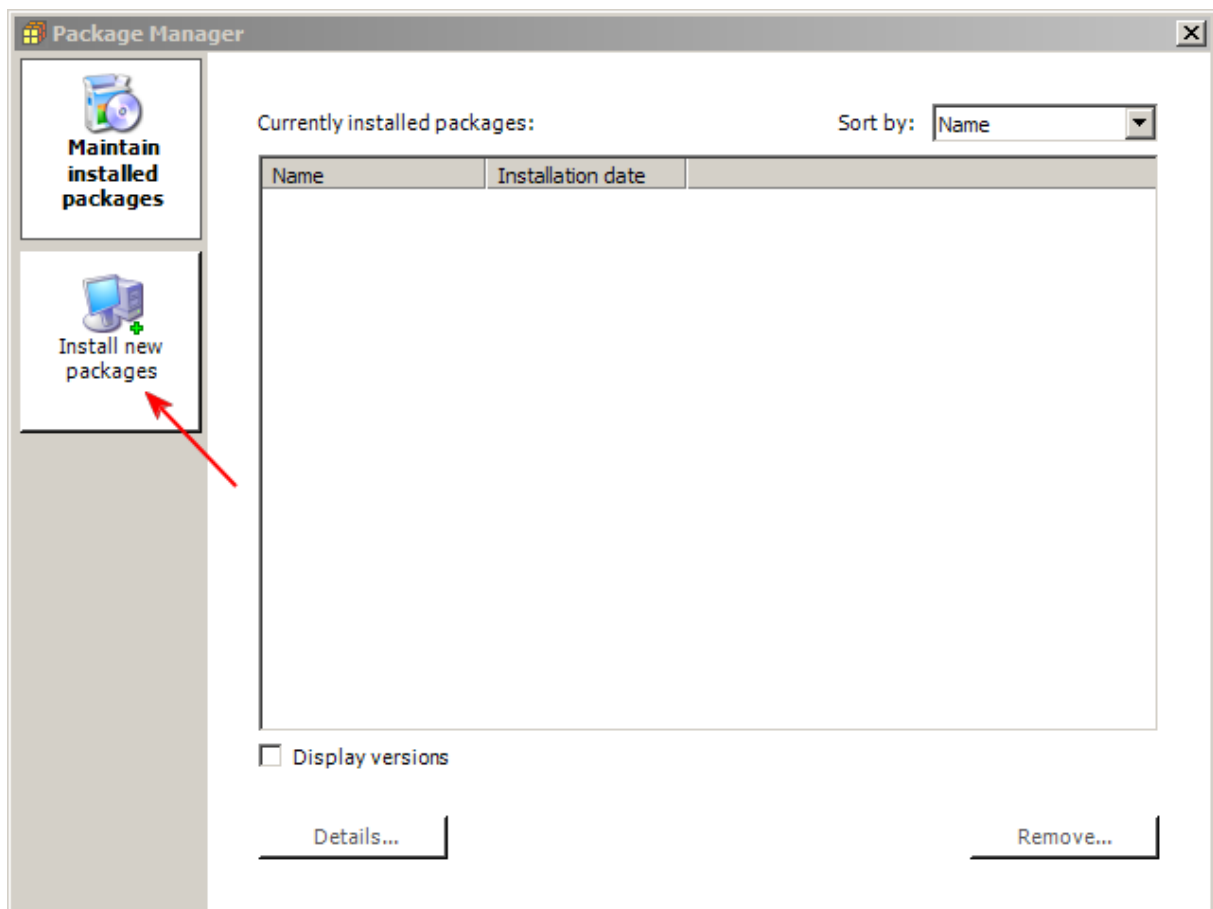
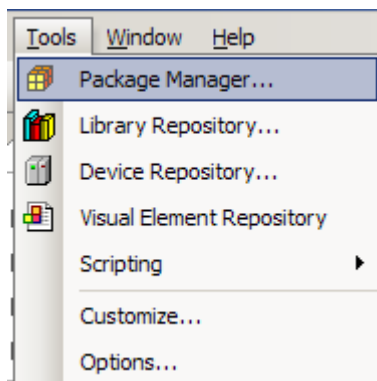
- | | |
|-----------------------|--|
| 1) *.bex | – The Executable runtime |
| 2) AUTOEXEC.BAT | – To Start the runtime (.bex) automatically after reboot |
| 3) CoDeSysControl.cfg | – configuration file of the runtime |
| 4) 3S.dat | – 3S license file |

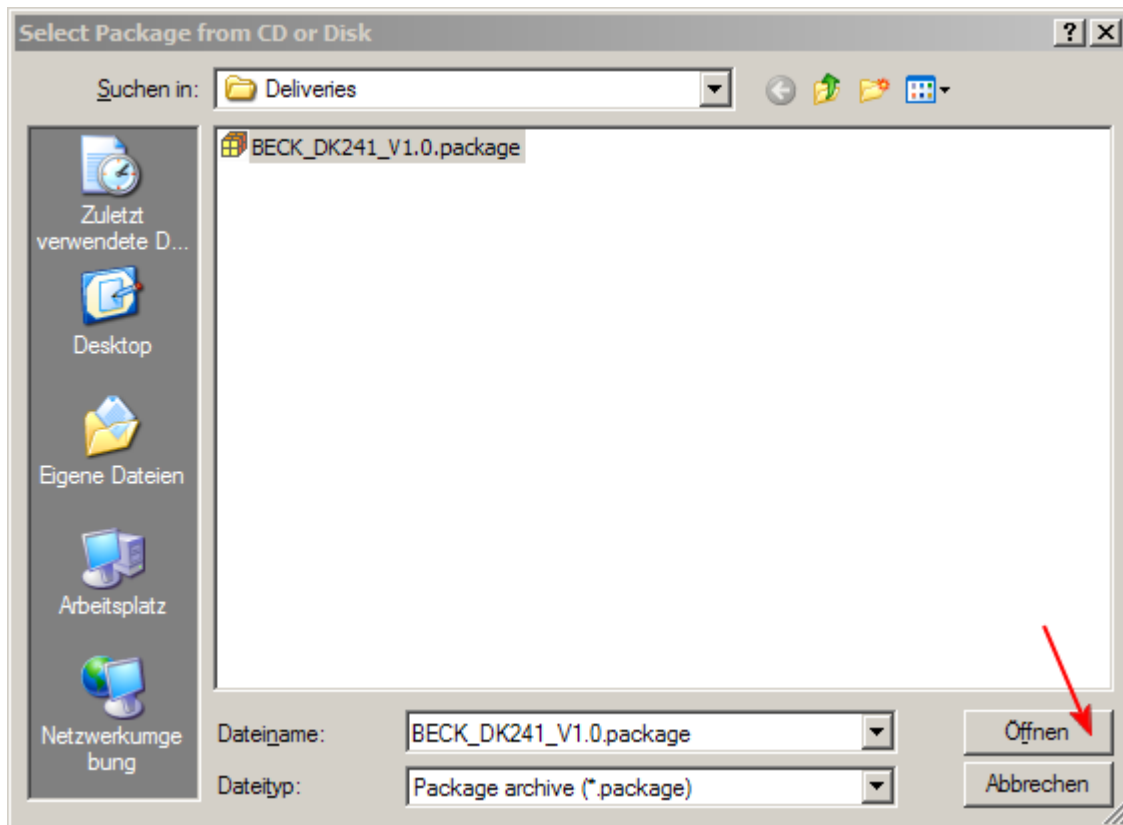
3.2 Installing the TSP - Setting up your CoDeSys V3 Programming System

The TSP (Target Support Package) is a collection of files which describe your IPC@CHIP® hardware and software to the CoDeSys Programming System. By installing the TSP, the collection of files is handed over to the CoDeSys environment.

The Installation of a TSP is very easy. You simply doubleclick on the package file **BECK_DK241_V1.0.package**. The CoDeSys V3 have to be installed previous, because the package manager which is part of the CoDeSys V3 installation is necessary to install packages.

It's also possible to install the package by starting the package manager directly. Within the CoDeSys V3 workbench select the **Tools->Package Manager** item.

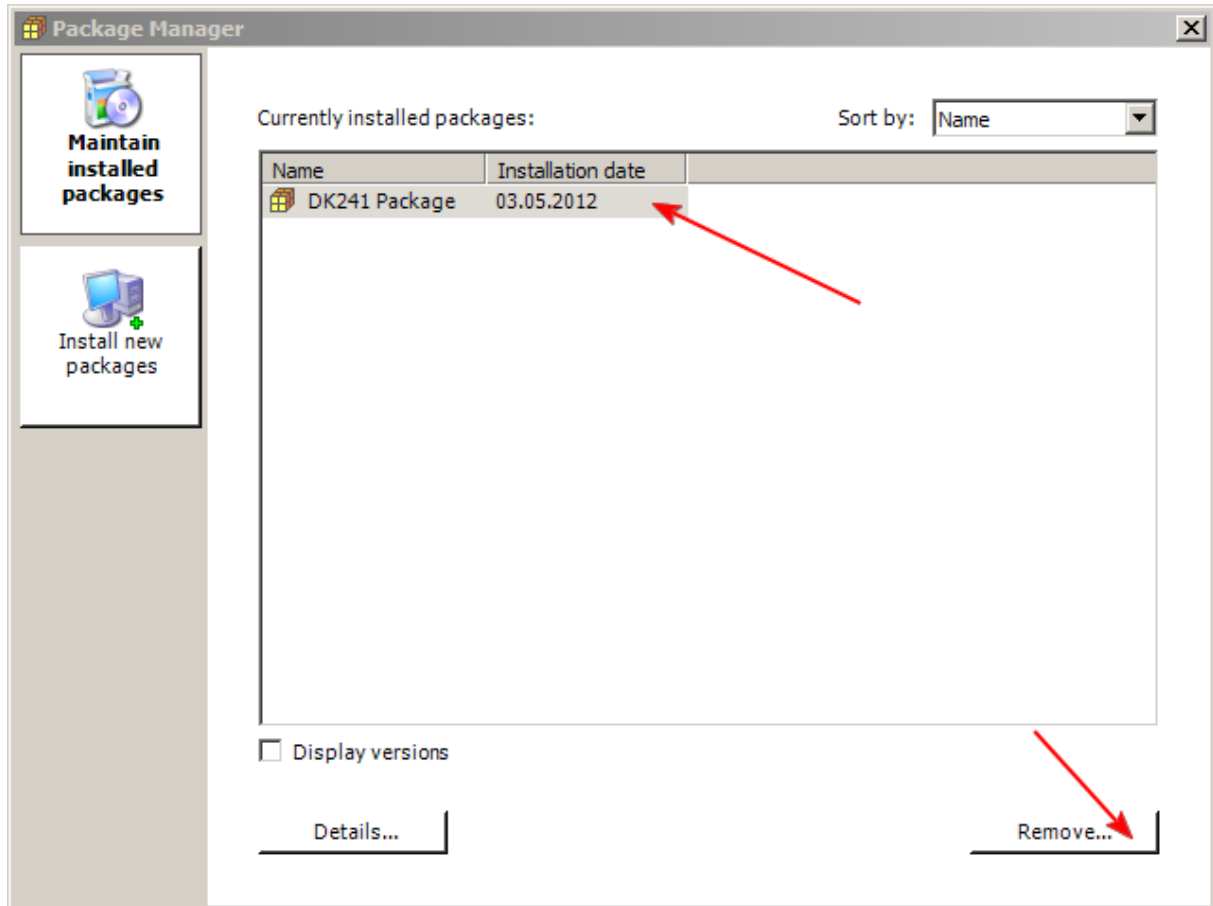




Click open to install the package.

3.3 Deinstalling a TSP

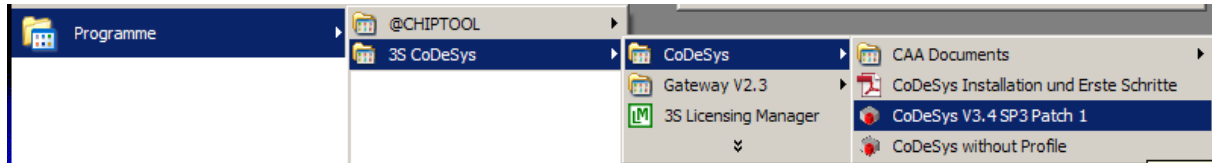
To uninstall a TSP you have to start the **Package Manager** like described before. Select the package you want to remove and click remove.



4 CoDeSys V3

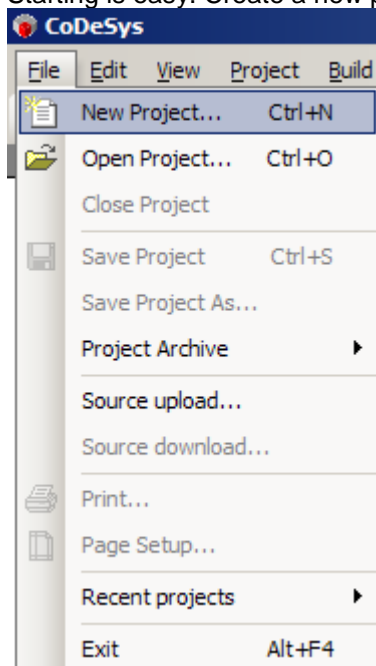
Start the CoDeSys programming system via the Start Menu:

Start → Programs → 3S CoDeSys → CoDeSys → CoDeSys V3.X

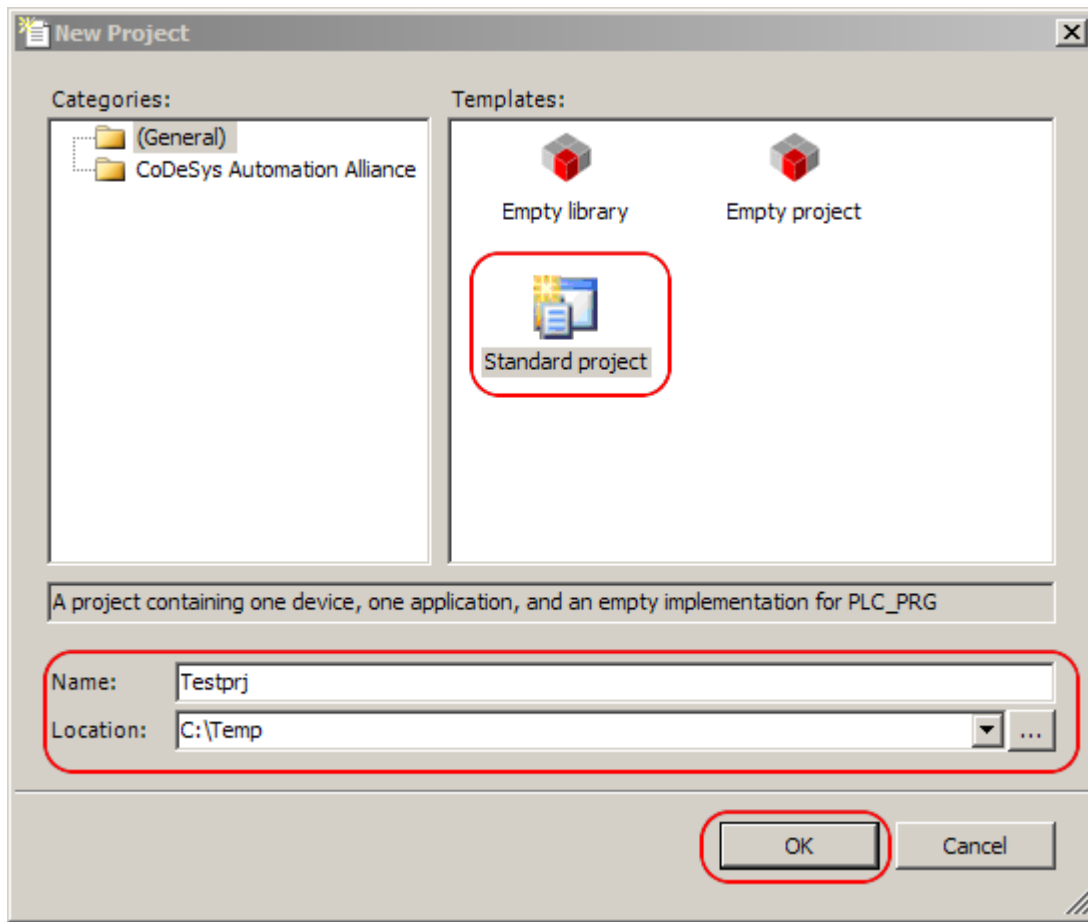


4.1 Creating a New Project

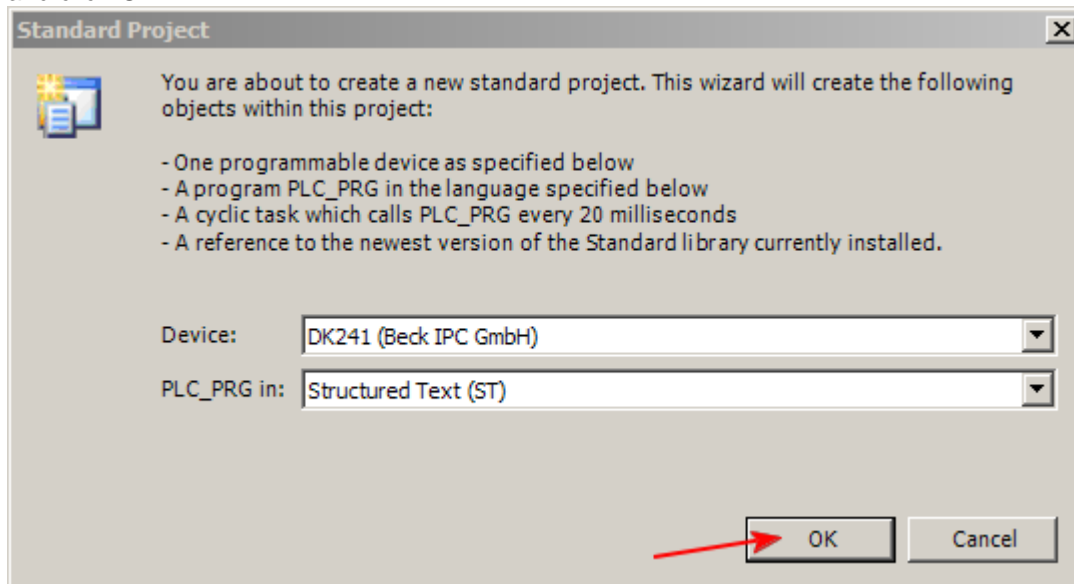
Starting is easy. Create a new project with the command File → New.project



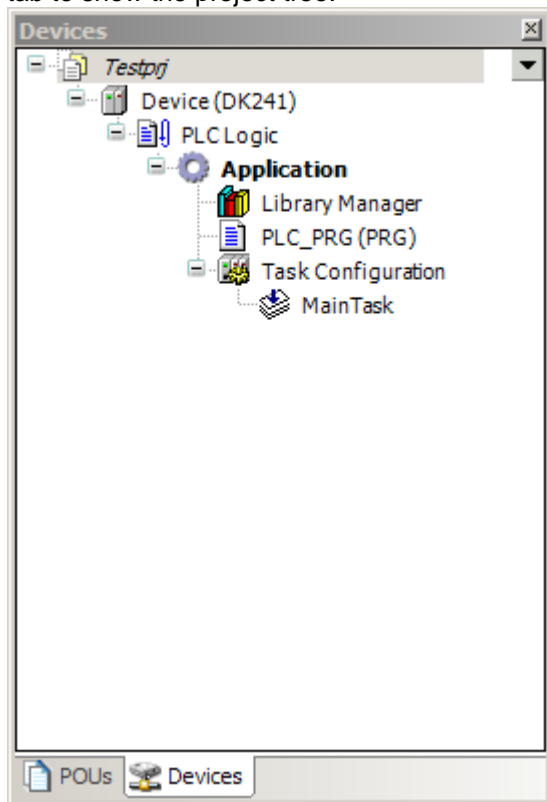
Select the **Standard project** in the following dialog insert a project name and the location, where the project file has to be stored. Then click **OK**.



For our first project please select the DK241 (Beck IPC GmbH) entry which matches your IEC Platform and click **OK**.



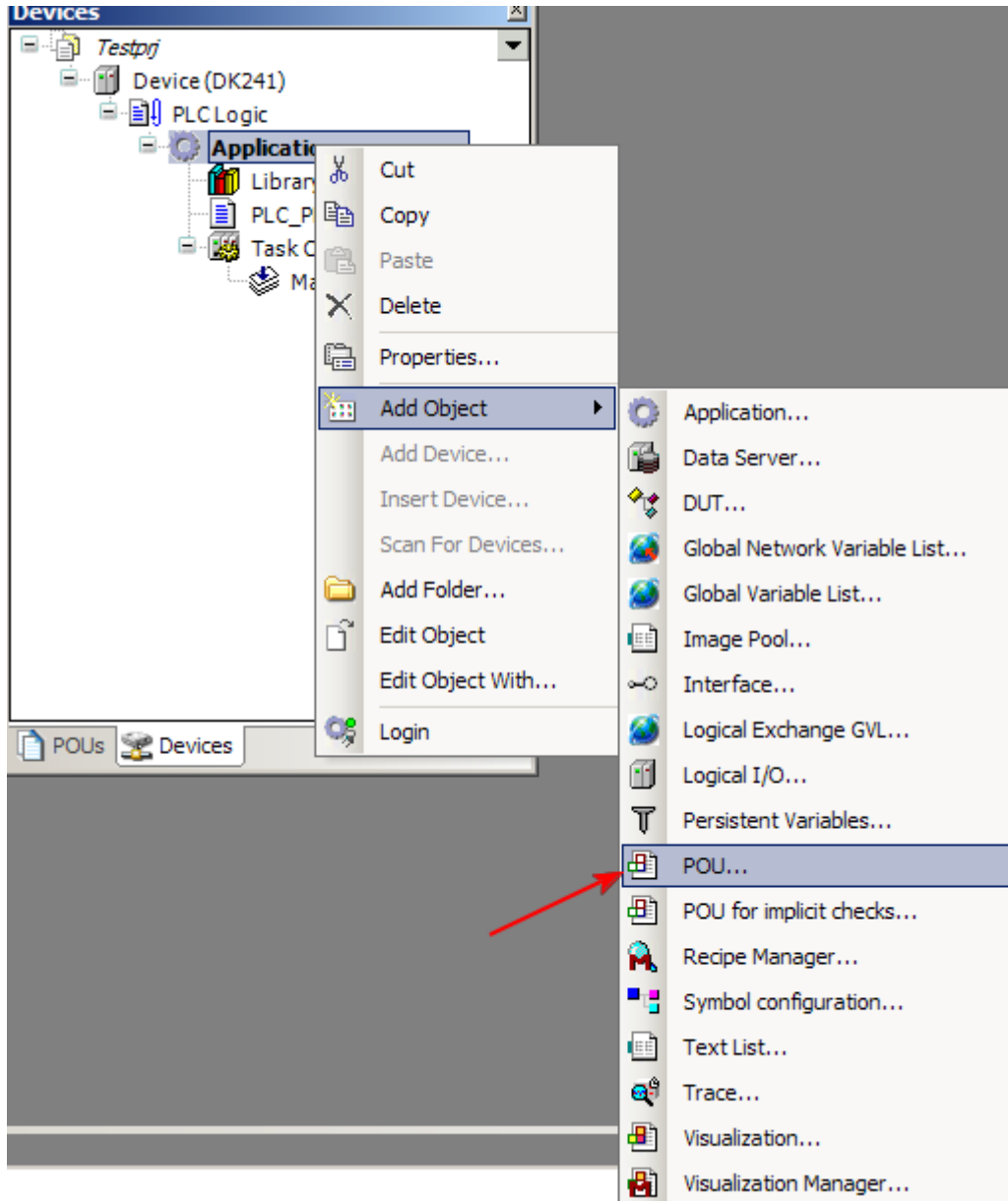
After click **OK** a new project is generated. The project appears in the POU view. Change to the Device tab to show the project tree.



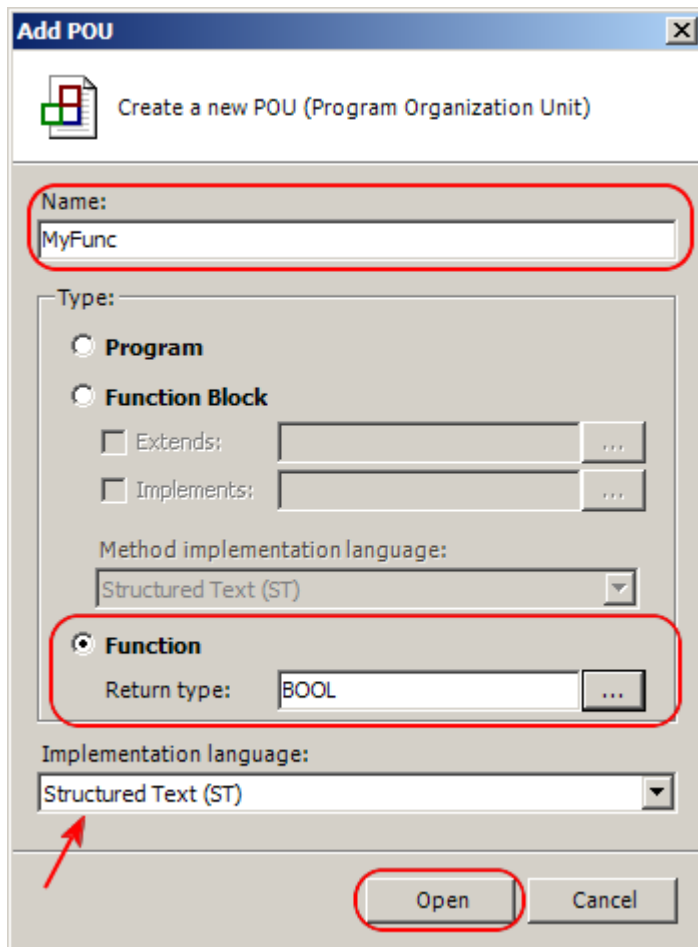
The created project has an empty PLC_PRG. The project can be compiled and downloaded to the target.

4.2 Create a POU

To create a new POU like a program, a function or a function block click with right mouse button on **Application**. Choose **Add Object->POU** in the context menu.



Insert a Name in the Name field, select the POU type (program, function block, function) and choose the programming language you want to use.



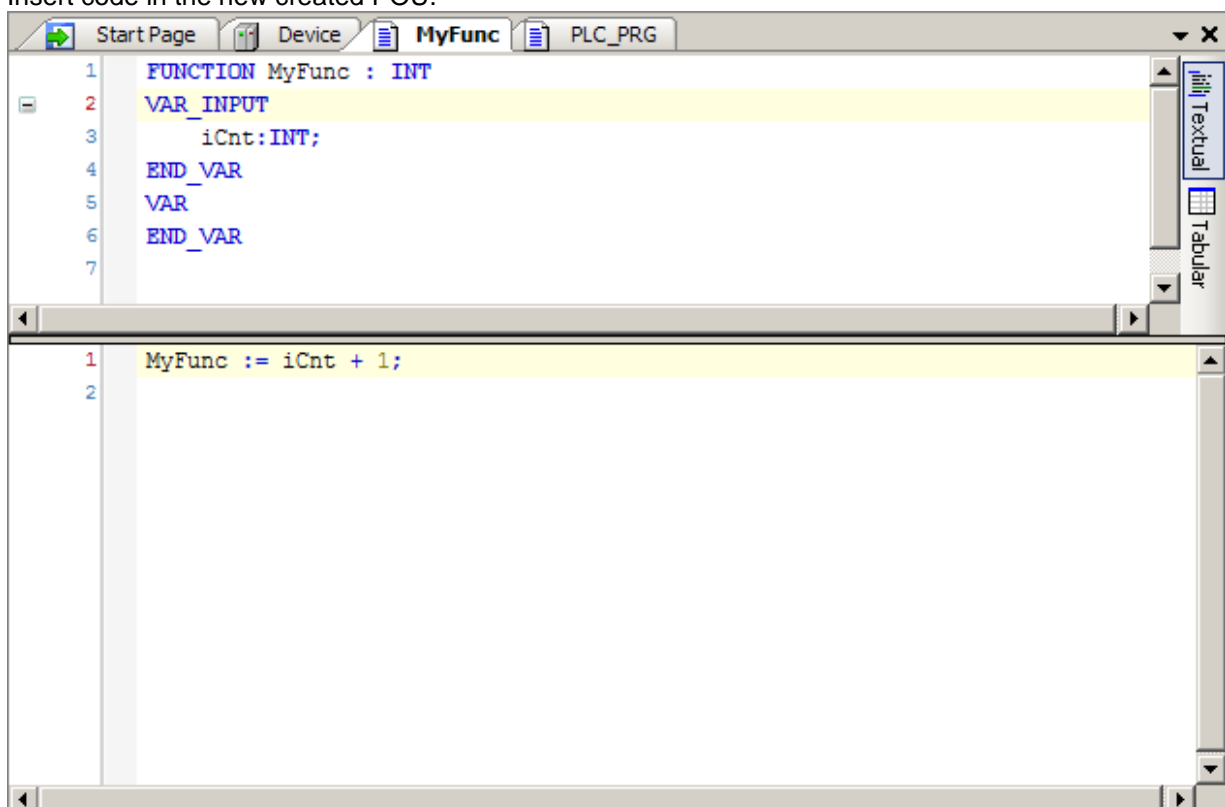
The 'Add POU' dialog box is shown with the following fields and options:

- Name:** MyFunc
- Type:**
 - ☐ Program
 - ☐ Function Block
 - ☐ Extends: []
 - ☐ Implements: []
 - Method implementation language: Structured Text (ST)
 - ☒ Function
 - Return type: BOOL
- Implementation language:** Structured Text (ST)
- Buttons:** Open, Cancel

Red circles highlight the Name field, the Function type option, the Return type field, and the Open button. A red arrow points to the Implementation language dropdown.

After click **Open** the new POU will be inserted in the project tree.

Insert code in the new created POU.

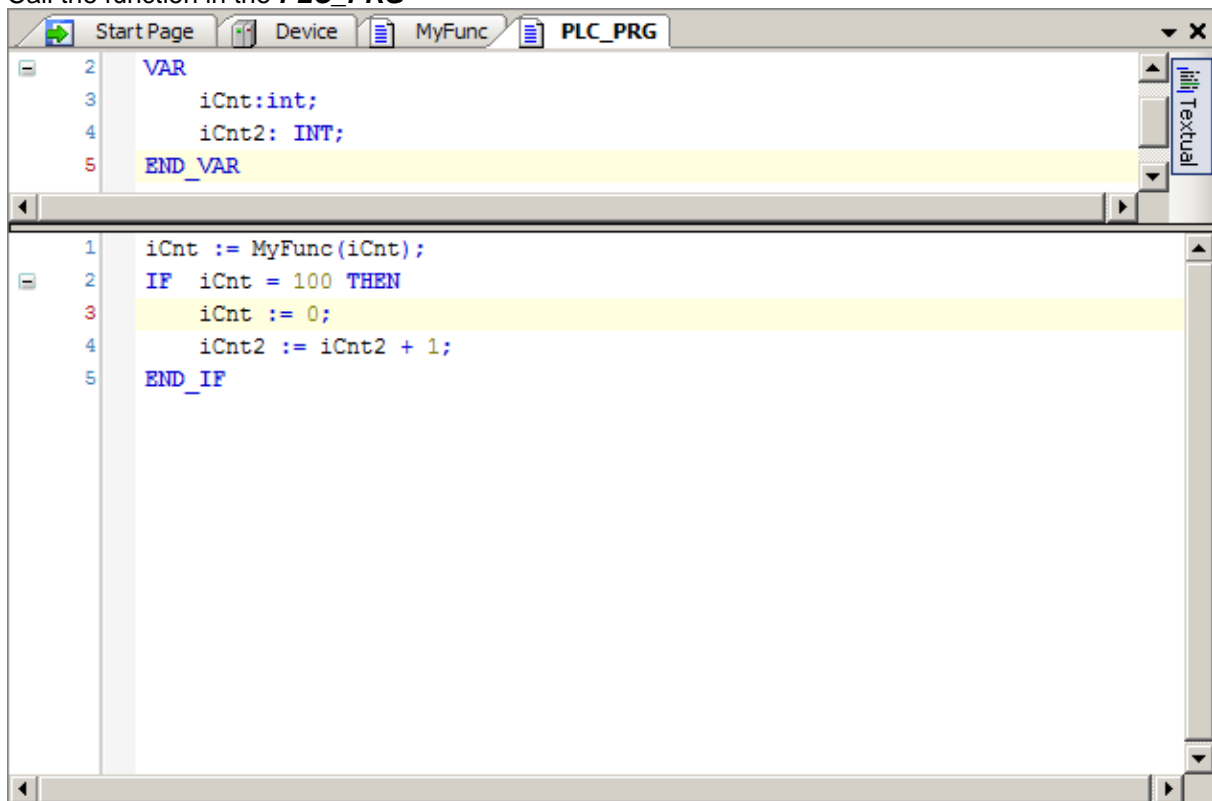


The code editor shows the following code for the new POU:

```
1 FUNCTION MyFunc : INT
2 VAR_INPUT
3     iCnt:INT;
4 END_VAR
5 VAR
6 END_VAR
7
1 MyFunc := iCnt + 1;
2
```

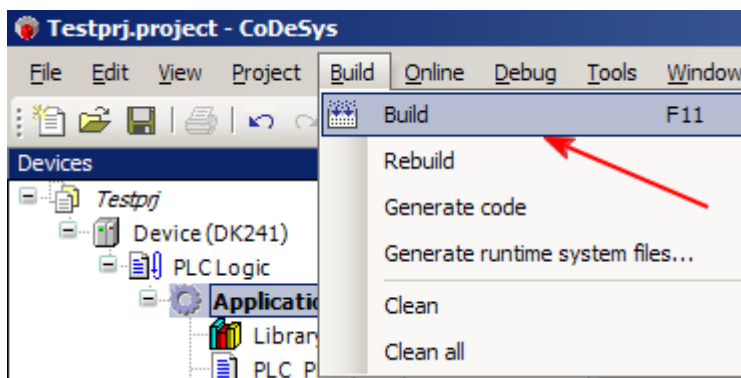
The editor has tabs for 'Start Page', 'Device', 'MyFunc', and 'PLC_PRG'. The 'MyFunc' tab is active. The code is color-coded: keywords in blue, variables in black, and comments in green.

Call the function in the **PLC_PRG**

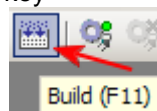


4.3 Build Project

Build the project with the menu command **Build→Rebuild**

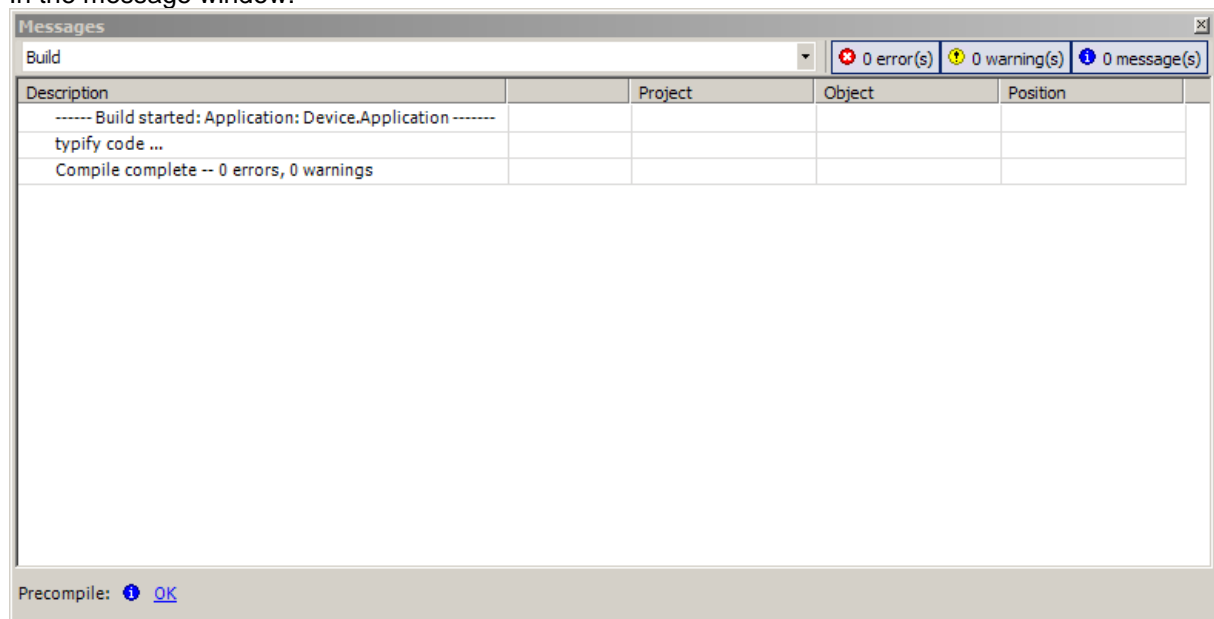


or with **<F11>** function key



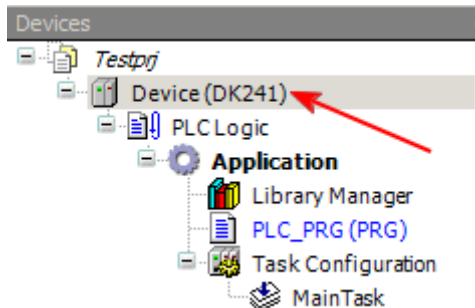
or with the build button

In the message window:



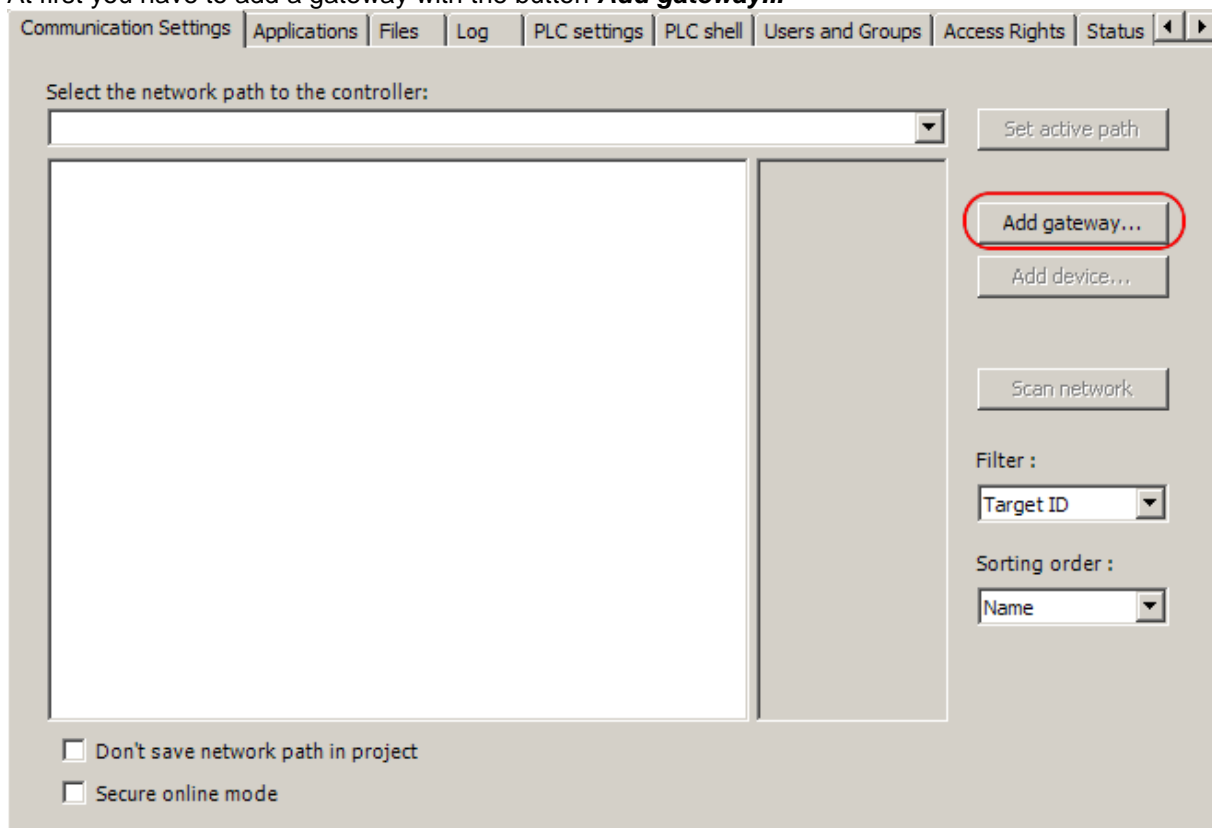
„**0 errors, 0 warnings**„ should be reported after the build procedure. If not, check that all the entries have been made correctly. The reported errors may also help.

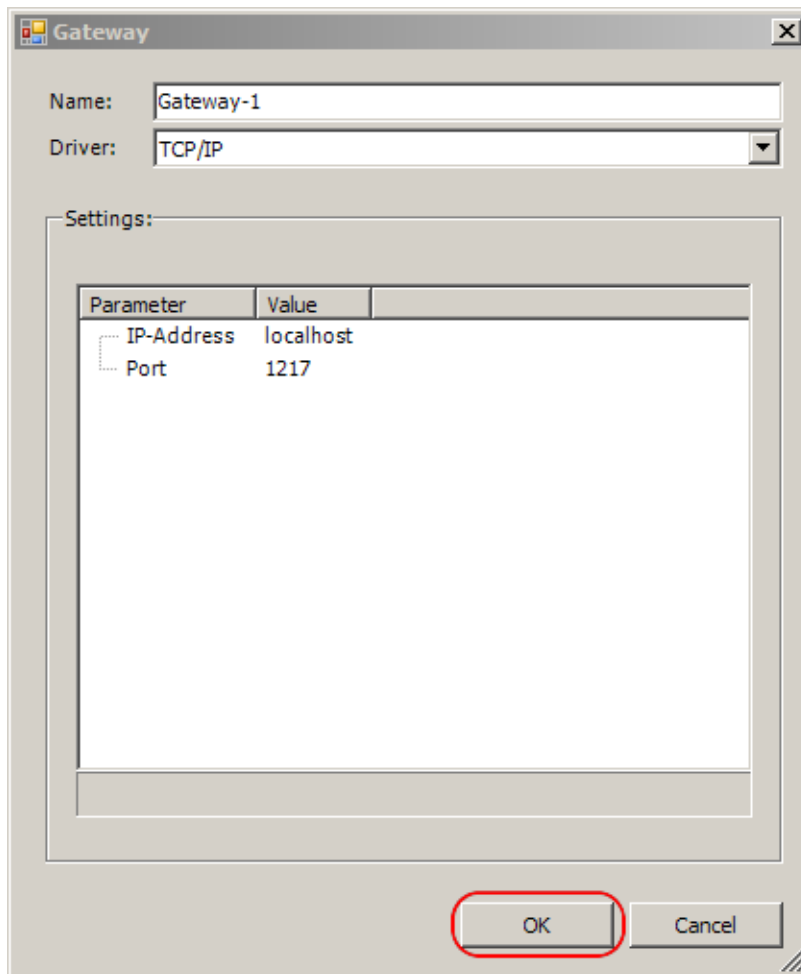
4.4 Establishing the Connection



Double click the **Device** in the project tree.

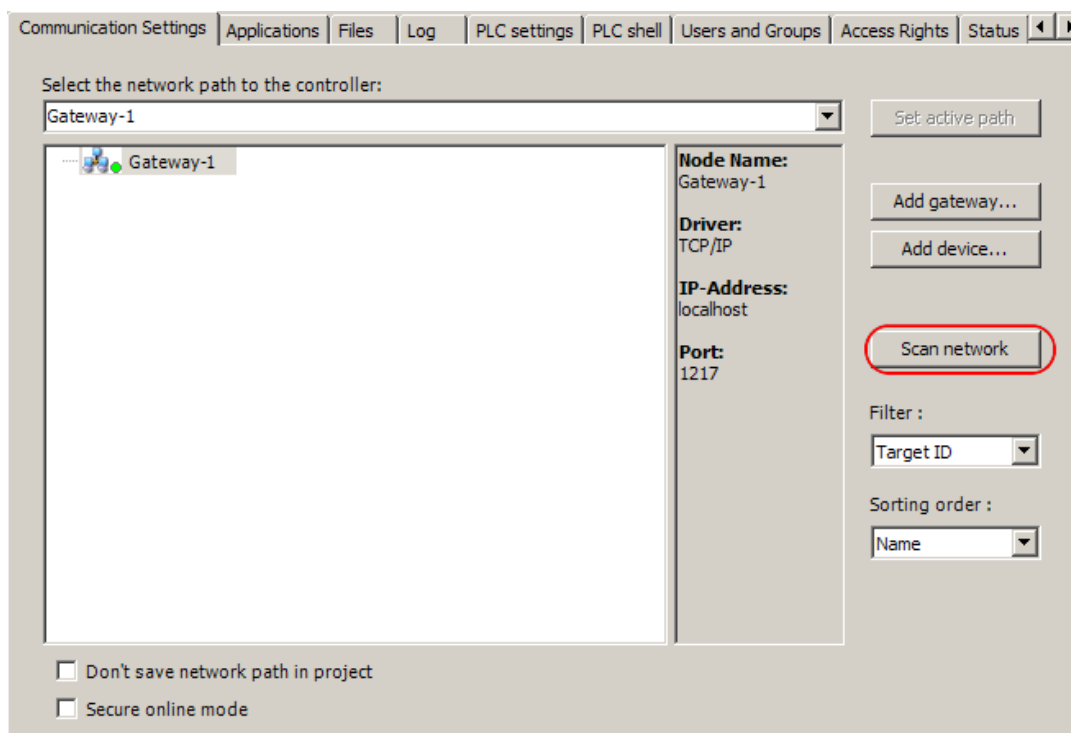
At first you have to add a gateway with the button **Add gateway...**



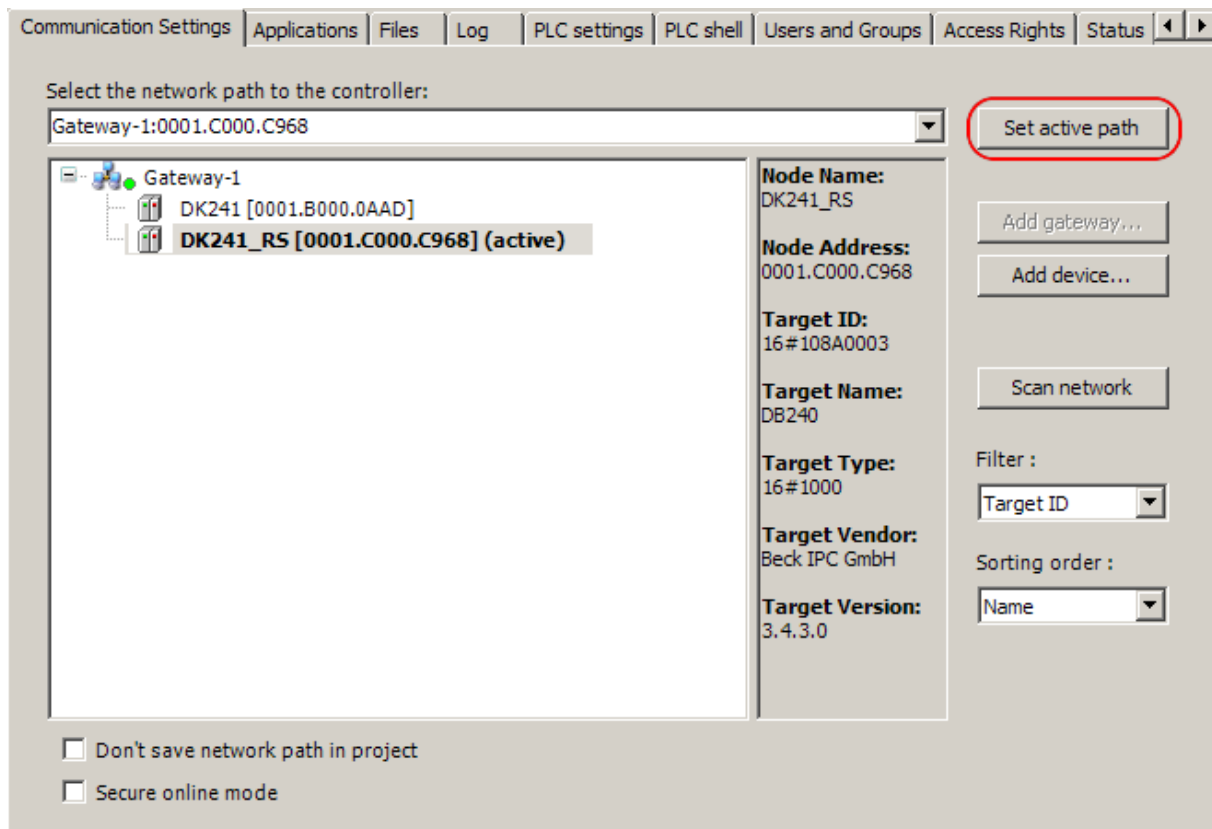


Click **OK** to create a Gateway with the standard settings.

Now it should appear a gateway entry in the dialog with a green dot. The green dot means, that the gateway is connected. If no gateway found, the dot is red. Select the **Scan network** button to find all devices connected to the gateway.

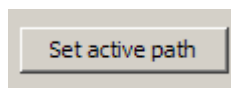


Below the gateway entry there should be listed all devices with a running CoDeSys V3 runtime which are connected to the network. The node address is a 12 byte hexadecimal number. The last two bytes are the subnet and device of the IP-Address.




Example of the address:

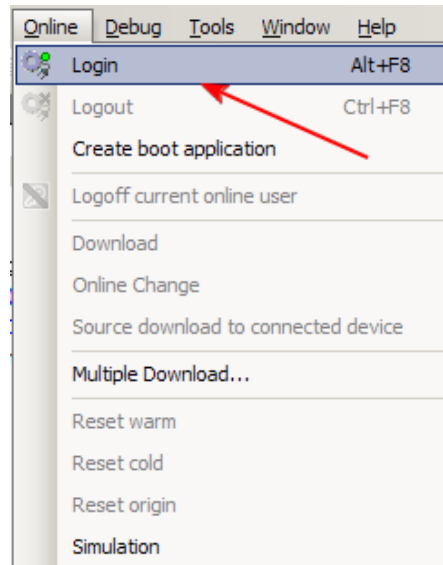
The IP-Address of the device is 192.168.201.104, the associated node number is 0001.C000.**C968** at which C9 is the subnet address = 201(decimal) and 68 the device address = 104(decimal) .



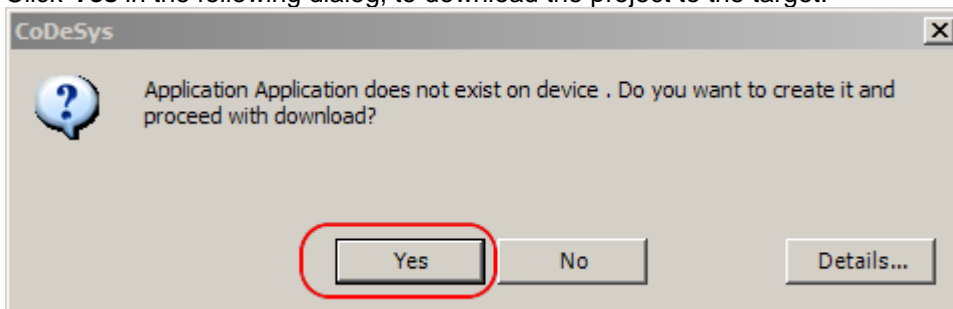
To choose the target as active mark the target and click on the right side of the dialog.

4.5 Online

To load the project into the device you have to go online. Therefore you click the login button  or choose in the menu the **Online->Login** item.

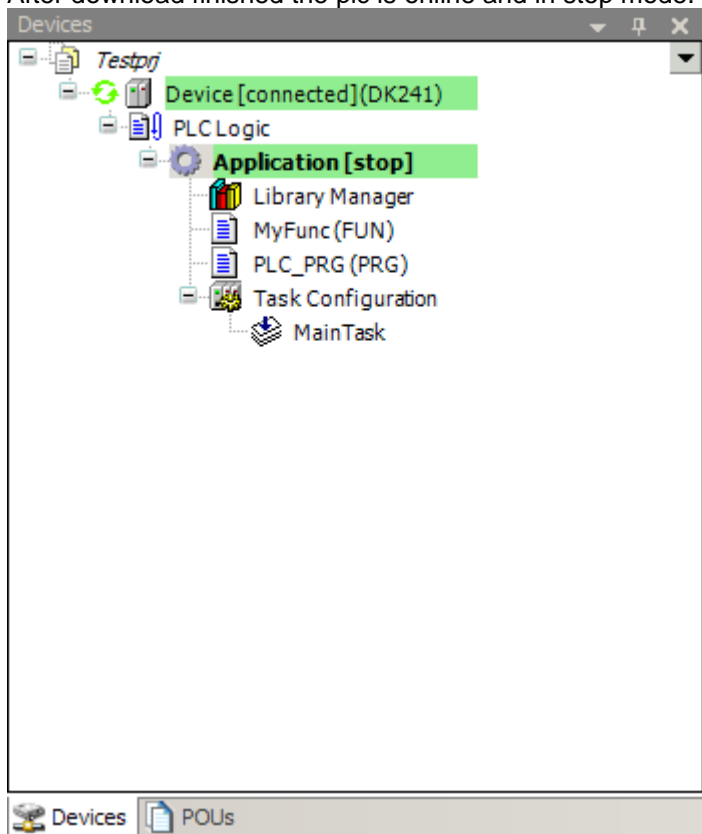


Click **Yes** in the following dialog, to download the project to the target.



The project will be loaded to device.

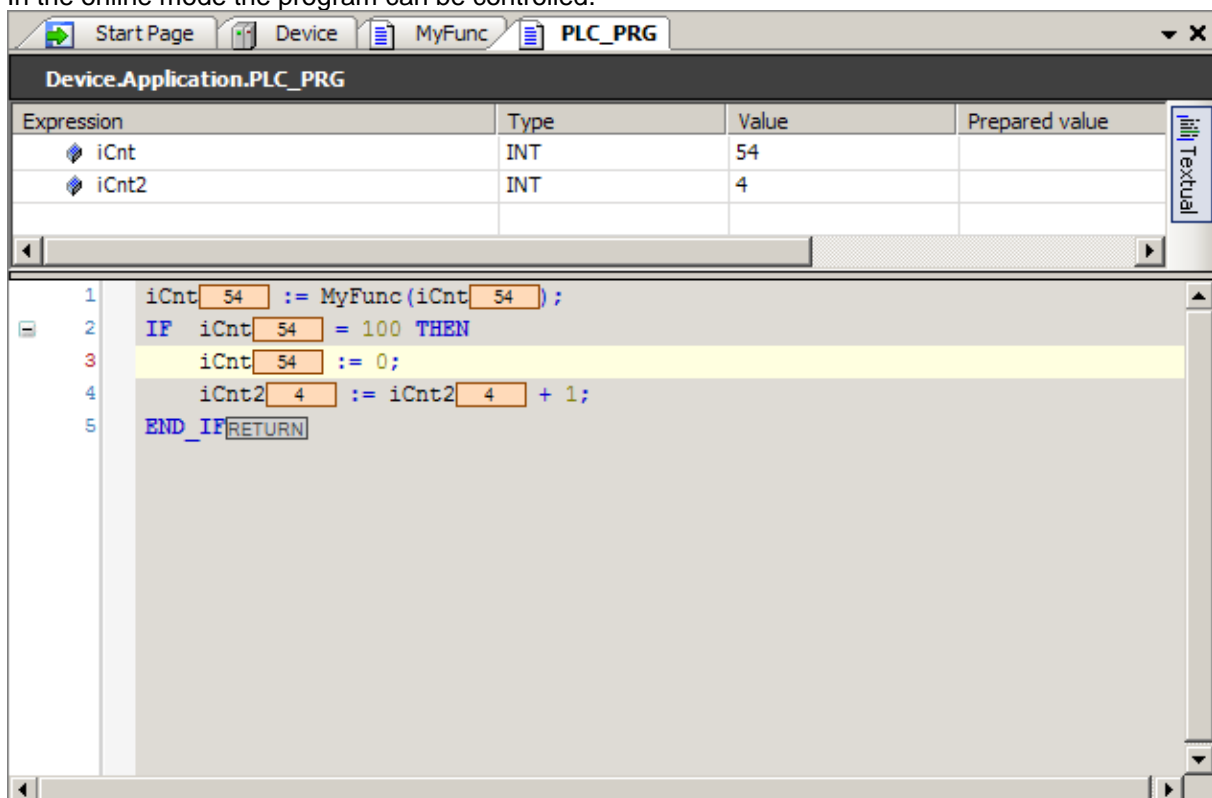
After download finished the plc is online and in stop mode.



To start to application press



In the online mode the program can be controlled.

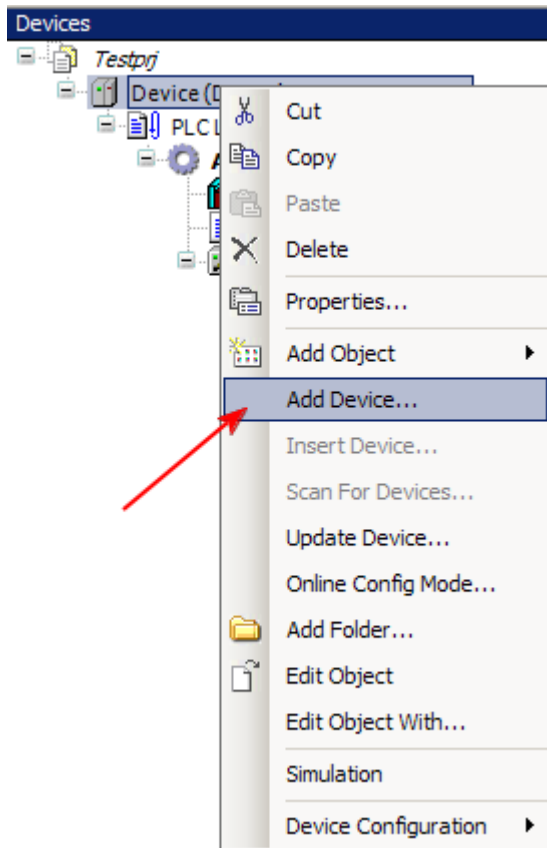


To stop the application press



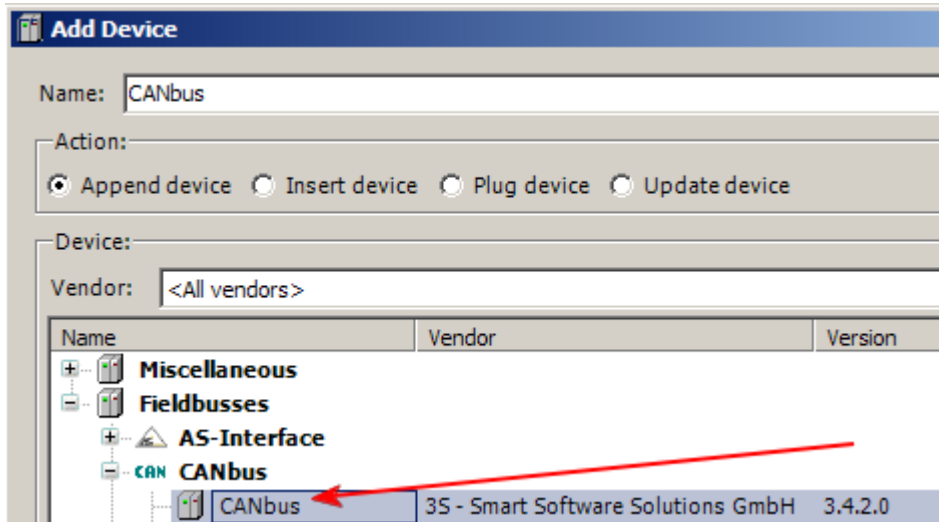
4.6 Add Hardware devices

To add a hardware device like an IO card or a field bus structure, make right click on the Device in the device tree.

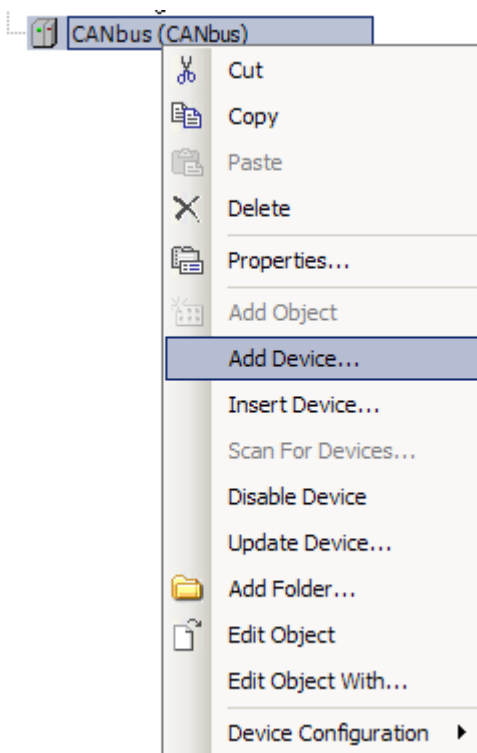


4.6.1 CANOpen

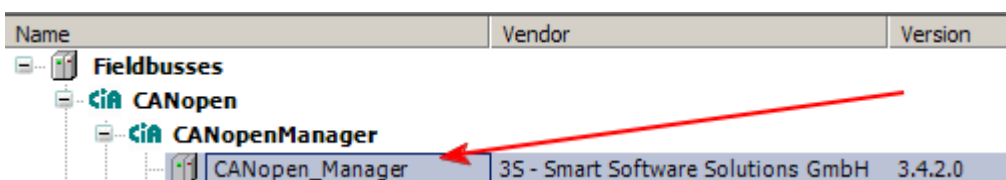
The CanOpen Interface will be supported by the DK241 target. To add the CAN bus select in the **Add Device** dialog **Fieldbusses->CANbus->CANbus**, and click **Add Device**.



Right click on the CANbus opens the context menu. Select **Add Device** to add a CANopen-Master.



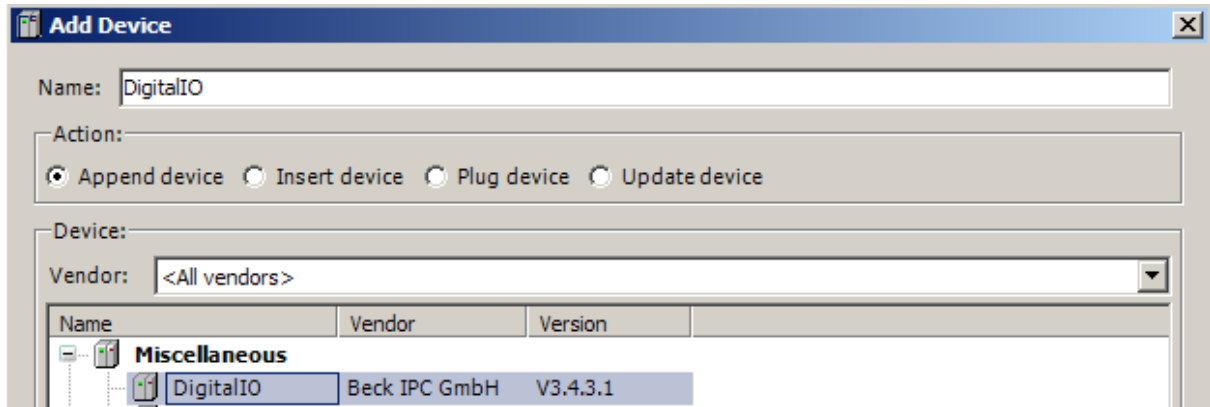
To add the CANopen Master select **CANopen_Manager** in the **Add Device** dialog and click **Add Device**.



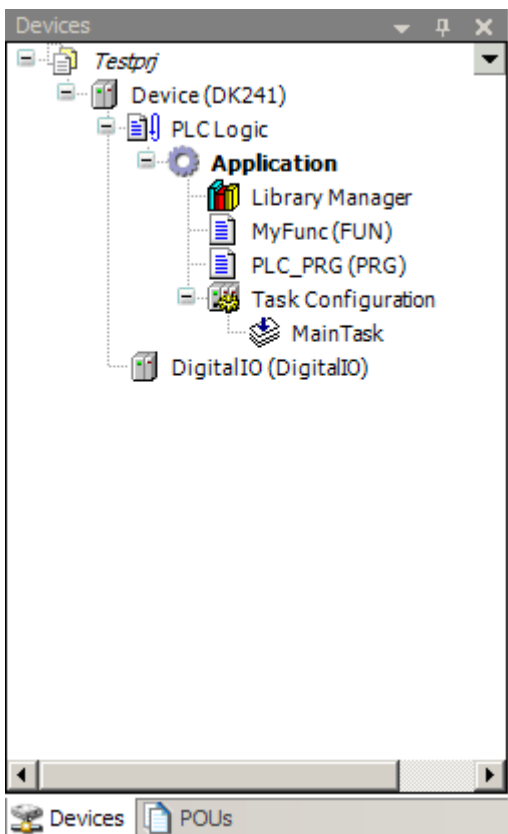
With **Add Device** on the **CANopen_Manager** there can be CANopen slaves added.

4.6.2 IO module

The IO device which supports the local digital IO on the DB240 can be added in the **Add Device** dialog via select **Miscellaneous->Digital_IOs**, and click **Add Device**.



The IO device will be inserted in the device tree.



To use the inputs and outputs there should variables mapped to the inputs and outputs. Double click on the IO device opens the settings dialog of the IO device. Select the tab Common.PCI I/O Mapping. Click in the variable field of input and insert a name of the variable. This variable can be used directly in a CoDeSys POU.

Common.PCI Configuration | Common.PCI I/O Mapping | Status | Information

Channels

Variable	Mapping	Channel	Address	Type	Default Value	Unit	Description
Inp1	...	in1	%IB0	BYTE	0		
Application		out1	%QB0	BYTE	0		

Reset mapping ☐ Always update variables

IEC Objects

Variable	Mapping	Type
DigitalIO_Instance		IoDrvDB240

= Create new variable = Map to existing variable

Common.DigIO Configuration | Common.DigIO I/O Mapping | Status | Information

Channels

Variable	Mapping	Channel	Address	Type	Default Value	Unit	Description
Inp1		in1	%IB0	BYTE	0		
Out1		out1	%QB0	BYTE	0		

Reset mapping ☐ Always update variables

IEC Objects

Variable	Mapping	Type
DigitalIO_Instance		IoDrvDB240

Start Page | Device | MyFunc | **PLC_PRG** | DigitalIO

```

1 PROGRAM PLC_PRG
2 VAR
3     iCnt: INT;
4     iCnt2: INT;
5     wType: WORD;
6 END_VAR

1 iCnt := MyFunc(iCnt);
2 IF iCnt = 100 THEN
3     iCnt := 0;
4     iCnt2 := iCnt2 + 1;
5 END_IF
6
7 Out1 := Inp1;
8

```


4.7 Create own IO Device

The CoDeSys-Runtime can be extended by own libraries or IO drivers which are in IEC. The detailed information how to create own devices respectively IO drivers will be explained in the online help.

Appendix A Additional Information

A.1 Where to find more information on the IPC@CHIP®

Check our website at <http://www.beck-ipc.com>

Here you will find application interfaces, tools, examples, new @CHIP-RTOS versions, etc.

Remember that the IPC@CHIP® is an evolving product. It makes sense to visit the IPC@CHIP® home page regularly.

A.2 Where to get support

There are two possible ways to get support:

1. To get support please use our newsgroup on the Internet:
You can visit the IPC@CHIP® forum at <http://www.beck-ipc.com/ipc/support/forum>. Here you will find a lot of answers to customer problems and you can also add your own ideas and questions. Please use email as the main way of communication. This not only reduces the time you spend on the phone, but also allows us to easily manage the questions and identify problem areas. Important or frequent questions and the corresponding answers are becoming part of our 'Frequently asked question list' that is published on the Internet.
2. To get support via email:
Please send your emails with your request to <mailto:support@beck-ipc.com>

You will find also a lot of answers in our Frequently Asked Questions documentation at <http://www.beck-ipc.com/ipc/support>.

A.3 Order numbers

The SC243 processors and any part of the DK241 Development Kit can be ordered separately. Please visit our Online Shop. The order numbers are:

Device	Ordernumber
SC243	563963
SC243-IEC	570736
DB240 Development board	563964
DK241 Development Kit	563965
PSE10 power supply	538934

End of this document